09, 98:170 C2C

PATENT
Attorney Docket No. 41679

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent No. 7,092,399

Confirmation No. 5639

Issued: August 15, 2006

Name of Patentee: David Cheriton

Patent Title: REDIRECTING MULTIPLE REQUESTS RECEIVED OVER A CONNECTION TO MULTIPLE SERVERS AND MERGING THE RESPONSES OVER THE CONNECTION

CERTIFICATE OF MAILING

I hereby certify that this paper is being deposited with the United States Postal Service on the date shown with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, Washington, D.C. 20231, on **September 4, 2006**.

9-4-2006

Kirk D. Williams, Esq.

REQUEST FOR CERTIFICATE OF CORRECTION OF PATENT FOR PATENT OFFICE MISTAKE (37 C.F.R. § 1.322)

Attn: Certificate of Correction Branch
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

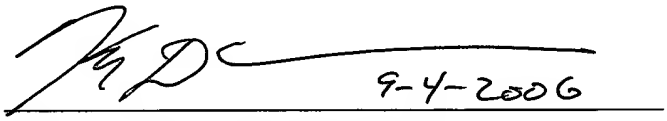
It is requested that a Certificate of Correction be issued to correct Office mistakes found the above-identified patent. Attached hereto is a Certificate of Correction which indicates the requested correction. For your convenience, also attached are copies of selected pages (a) from the issued patent with errors highlighted, (b) from the original application as filed October 16, 2001 and (c) Amendment A filed December 23, 2005, with the correct text/instructions.

In re US Patent No. 7,092,399

It is believed that there is no charge for this request because applicant or applicants were not responsible for such error, as will be apparent upon a comparison of the issued patent with the application as filed or amended. However, the Assistant Commissioner is hereby authorized to charge any fee that may be required to Deposit Account No. 501430.

Respectfully submitted,
The Law Office of Kirk D. Williams

Date: September 4, 2006

By 

Kirk D. Williams, Reg. No. 42,229
One of the Attorneys for Applicants
CUSTOMER NUMBER 26327
The Law Office of Kirk D. Williams
1234 S. OGDEN ST., Denver, CO 80210
303-282-0151 (telephone), 303-778-0748 (facsimile)

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 7,092,399

DATED : August 15, 2006

INVENTOR(S) : David Cheriton

It is certified that error(s) appear in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Col. 8, line 25, replace "The invention claimed" with -- What is claimed --

Col. 8, line 66, replace "plurality of indications" with -- plurality of splice indications --

Col. 10, line 13, replace "servers updating" with -- servers, updating --

MAILING ADDRESS OF SENDER:
Kirk D. Williams, Reg. No. 42,229
Customer No. 26327
The Law Office of Kirk D. Williams
1234 S. Ogden Street, Denver, CO 80210

PATENT NO. 7,092,399
No. of additional copies

⇨ NONE (0)

SEP 13 2006

315 to one of the servers is being allocated to service the particular request. The particular request is forwarded by server interface 312 over one of the established server connections 315, establishes a new connection, or via some other method. In one embodiment, a splicer token is also sent over the selected server connection 315. This process is repeated for multiple requests from the same clients, as well as for multiple clients.

Server interface 312 of switch with splicer 300 receives the responses over server connections 315 (or via some other mechanism). Client address translator 314 redirects the response by typically performing a network address translation and possibly modifying or adding a sequence number corresponding to that of the particular one of the client connections 305, such that a client can receive responses from its multiple requests over a single logical connection (e.g., TCP connection). Client interface 306 forwards the packets to the clients. In one embodiment, client address translator 314 further receives indications of splicer responses, and updates one or more data structures 310.

FIG. 3B illustrates one embodiment of a connection data structure 320 used in one embodiment to maintain a set of pre-established connections to servers. In one embodiment, connection data structure 320 is in the form of an array 330 with an entry for each of the n servers, with a linked list pointer to arrays of connection identifiers 331–339 which indicate available connections to a particular server.

FIG. 3C illustrates one embodiment of a splicer data structure 340 used in one embodiment to maintain information about outstanding requests sent to a server. In one embodiment, splicer data structure 340 is in the form of an array 350 with an entry for each of the available servers, with a linked list entry to an array 351–359 of information of outstanding requests and client connections. In one embodiment, a connection identifier 351A–359A is maintained for a server connection being used, along with a corresponding client address 351B–359B, and a sequence number 351C–359C for use in splicing and sending multiple responses over a single connection to each of the clients.

The processing by one embodiment is further explained in relation to the flow diagram of FIG. 4A. Processing begins at process block 400, and proceeds to process block 402, wherein a set of connections are pre-established to a bank of servers. Next, in process block 404, the connection and splicer data structures are initialized to indicated the establishment of these connections and other housekeeping data.

A request is received from a client in process block 406, and a server to which to respond to the request is determined in process block 408. If, as determined in process block 410, that a connection is not established to the determined server, then in process block 412, one or more connections are established to the server, and one or more data structures are updated to reflect the new connection or connections. Next, in process block 414, a particular connection to the determined server to use is selected and the one or more data structures are updated. In process block 416, the request is forwarded over the selected connection; and in process block 418, a splicer token is sent to the determined server over the selected connection. Processing returns to process block 406 to process more client requests.

The processing of one embodiment is further explained in relation to the flow diagram of FIG. 4B. Processing begins at process block 440, and proceeds to process block 442, wherein a response is received from a server. Next, as determined in process block 444, if the response is a splicer token response, then in process block 446, the one or more data structures are updated in process block 446, such as, but not limited to indicating that the connection to the responding server is no longer in use. Otherwise, in process block 448, the response is redirected to the originating client over the single connection. In one embodiment, the data portion of the response is included in one more packets with new a header information indicating the address of the client and/or an appropriate sequence number or numbers for the single connection the client. In process block 450, the redirected response is sent to the client over the single connection. Processing returns to process block 442 to receive and process more server responses.

In view of the many possible embodiments to which the principles of our invention may be applied, it will be appreciated that the embodiments and aspects thereof described herein with respect to the drawings/figures are only illustrative and should not be taken as limiting the scope of the invention. For example and as would be apparent to one skilled in the art, many of the process block operations can be re-ordered to be performed before, after, or substantially concurrent with other operations. Also, many different forms of data structures could be used in various embodiments. The invention as described herein contemplates all such embodiments as may come within the scope of the following claims and equivalents thereof.

The invention claimed is:

1. A switch comprising:

a memory configured to store connection information;

a server address translator configured to receive a plurality of requests from a client over a single connection, to reference the memory to determine a plurality of servers to service said received plurality of requests; and to redirect said received plurality of requests to said determined plurality of servers; wherein the server address translator is configured to indicate a splice token request to a current server of said determined plurality of servers in response to a redirection of said requests from the client to a server different from the current server of said determined plurality of servers; and

a client address translator configured to receive a plurality of responses, including a plurality of splice token responses to said splice token requests, from said determined plurality of servers; to organize said received plurality of responses into a stream of packets; and to forward said stream of packets over the connection to the client; wherein the client address translator is configured update the memory in response to said splice token response in order to identify whether or not to switch to a different server than a current server of said determined servers for said responses.

2. The switch of claim 1, wherein said each of said splice token responses is sent in a packet separate from said response(s) received from the current server which said splice token response corresponds.

3. A method comprising:

receiving a plurality of requests from a client over a single Transmission Control Protocol (TCP) connection;

redirecting the plurality of requests to a plurality of servers;

receiving a plurality of responses from the plurality of servers;

organizing the plurality of the responses into a stream of packets;

sending the stream of packets to the client over the single connection;

sending a plurality of indications to the plurality of servers;

receiving a plurality of splice indication responses from the plurality of servers; and

wherein said organizing the plurality of the responses includes referencing the plurality of splice indication responses.

4. The method of claim 3, further comprising updating the memory or a second memory in response to receiving the plurality of splice indication responses.

5. The method of claim 3, wherein said each of said splice token responses is received in a packet separate from said response(s) received from the server of said plurality of servers which said splice token response corresponds.

6. A method comprising:

receiving a first request over a connection from a client;

redirecting the first request to a first server;

receiving a first response to the first request from the first server;

forwarding the first response over the connection to the client;

receiving a second request over the connection from the client after said redirecting the first request to the first server;

in response to identifying that the second request should be sent to a second server different from the first server, sending a first splice token to the first server to indicate the redirection of requests from the client;

redirecting the second request to the second server;

receiving a second response to the second request from the second server; and

forwarding the second response over the connection to the client.

7. The method of claim 6, further comprising: comprising receiving a first splice token response from the first server.

8. The method of claim 7, further comprising updating a memory for storing splicer data in response to said receiving the first splice token response in order to identify whether or not to switch to the second server for the second response.

9. The method of claim 8, wherein the splicer data indicates an address of the client.

10. The method of claim 8, wherein the splicer data indicates a sequence number for a set of packets received from the client.

11. The method of claim 6, further comprising selecting the first server from a set of server identifiers maintained in a memory configured to store connection information.

12. The method of claim 6, wherein the second request is received prior to receiving and forwarding the first response to the client.

13. A method comprising:

establishing a set of connections to a plurality of servers;

maintaining an indication of the set of connections;

receiving a first request over a Transmission Control Protocol (TCP) connection from a client;

referencing the indication to determine a first one of the plurality of servers;

redirecting the first request to the first one of the plurality of servers;

receiving a first response to the first request from the first one of the plurality of servers;

receiving a second request over the connection from the client after said redirecting the first request to the first one of the plurality of servers;

referencing the indication to determine a second one of the plurality of servers;

sending a splice token request to the first one of the plurality of servers after redirecting the first request to the first one of the plurality of servers, and in response to said determination of the second one of the plurality of servers;

redirecting the second request to the second one of the plurality of servers;

in response to receiving a splice token response based on the splice token request from the first one of the plurality of servers updating the indication in response to receiving the splice token response to identify to receive responses from the second one of the plurality of servers;

receiving a second response to the second request from the second one of the plurality of servers; and

organizing the first and second responses into a stream of packets.

14. An apparatus comprising:

means for receiving a plurality of requests from a client over a single connection;

means for redirecting the plurality of requests to a plurality of servers;

means for receiving a plurality of responses from the plurality of servers;

means for organizing the plurality of the responses into a stream of packets;

means for sending the stream of packets to the client over the single connection;

means for sending a plurality of splice indications to the plurality of servers;

means for receiving a plurality of splice token responses from the plurality of servers; and

wherein said means for organizing the plurality of the responses includes means for referencing the plurality of splice token responses.

15. One or more computer-readable media containing computer-executable instructions for performing operations comprising:

receiving a first request over a connection from a client;

redirecting the first request to a first server;

receiving a first response to the first request from the first server;

forwarding the first response over the connection to the client;

receiving a second request over the connection from the client after said redirecting the first request to the first server;

in response to identifying that the second request should be sent to a second server different from the first server, sending a first splice token to the first server to indicate the redirection of requests from the client;

redirecting the second request to the second server;

receiving a second response to the second request from the second server; and

forwarding the second response over the connection to the client.

16. The computer-readable media of claim 15, wherein said operations further comprise receiving a first splice token response from the first server.

From Patent Application Pg 10-11-2001

CLAIMS

Col 8, line 25

What is claimed is:

1. A switch comprising:

a memory configured to store connection information;

a server address translator configured to receive a plurality of requests from a client over a single connection, to reference the memory to determine a plurality of servers to service said received plurality of requests; and to redirect said received plurality of requests to said determined plurality of servers; and

a client address translator configured to receive a plurality of responses from said determined plurality of servers; to organize said received plurality of responses into a stream of packets; and to forward said stream of packets over the connection to the client.

2. The switch of claim 1, wherein the server address translator is further configured to send a plurality of splicer tokens to said determined plurality of servers; and wherein the client address translator is further configured to receive a plurality of splicer token responses; and to update the memory in response to said receipt of the plurality of splicer token responses.

From Amendment A filed 12-23-2005

Claim 4 (currently amended): ~~The method of claim 3, further~~ A method comprising:

receiving a plurality of requests from a client over a single Transmission Control Protocol (TCP) connection;

redirecting the plurality of requests to a plurality of servers;

receiving a plurality of responses from the plurality of servers;

organizing the plurality of the responses into a stream of packets;

sending the stream of packets to the client over the single connection; Col 8, line 66

sending a plurality of ~~splicer~~ splice indications to the plurality of servers; ~~and~~

receiving a plurality of ~~splicer~~ splice indication responses from the plurality of servers; and

wherein said organizing the plurality of the responses includes referencing the plurality of ~~splicer~~ splice indication responses.

Claim 5 (currently amended): The method of claim 4, further comprising updating the memory or a second memory in response to receiving the plurality of ~~splicer~~ splice indication responses.

Claims 6-8 (canceled)

From Amendment A filed 12-23-2005

Renumbered as claim 13

Claim ~~16~~ (currently amended): A method comprising:

establishing a set of connections to a plurality of servers;

maintaining an indication of the set of connections;

receiving a first request over a Transmission Control Protocol (TCP) connection from a client;

referencing the indication to determine a first one of the plurality of servers;

redirecting the first request to the first one of the plurality of servers;

receiving a first response to the first request from the first one of the plurality of servers;

receiving a second request over the connection from the client after said redirecting the first request to the first one of the plurality of servers; ~~before said receiving the first response;~~

referencing the indication to determine a second one of the plurality of servers;

sending a splice token request to the first one of the plurality of servers after redirecting the first request to the first one of the plurality of servers, and in response to said determination of the second one of the plurality of servers;

redirecting the second request to the second one of the plurality of servers;

in response to receiving a splice token response based on the splice token request from the first one of the plurality of servers, updating the indication in response to receiving the splice token response to identify to receive responses from the second one of the plurality of servers;

Cl 10, to 13

receiving a second response to the second request from the second one of the plurality of servers; and

organizing the first and second responses into a stream of packets.